Gateway

Vol. 3, No. 10
Jan. 9, 1986

The ARRL Packet-Radio Newsletter

NEW PACKET RULES FOR AUSTRALIA

The Department of Communications (DOC) in Australia released their new packet regulations on October 20, 1986. The (abridged) rules are as follow:

(1) Novice Amateur Stations shall not use the packet radio transmission mode of operation.
(2) Amateur stations utilising "packet radio" must conform to the general technical parameters and conditions applying to the Amateur Service.
(3) Each "packet" shall contain the originating station's identification, that of the destination station and the station transmitting (if different from the originating station).
(4) Amateur stations employing "packet" in an unattended operating configuration shall be fitted with:

(i) a timer to cause automatic shut-down of the station transmitter after 10 minutes of uninterrupted transmission.
(ii) a fail-safe facility to prevent the station transmitter operating due to equipment malfunction.

(5) An amateur station shall not retransmit a "packet" signal in any amateur band that the originating station is not authorized to use.
(6) Amateur stations when utilising the packet radio transmission mode shall not be connected to the switched telephone network.

IMPORTANT NOTES:

(A) Any protocol may be used for "packet" transmission provided it meets the identification requirements stipulated in (3) above.
(B) The use of store/forward packet techniques by stations in the Amateur Service is permitted.
(C) Amateur licensees employing the packet radio mode of transmission are reminded that they are responsible for ensuring that third party traffic conditions are met.
(D) Providing the conditions stipulated in (4) above are met, amateur stations may operate in an unattended configuration when utilising the packet radio transmission mode.
(E) Packet repeater stations must comply with the conditions applicable to repeater stations and those conditions outlined above.

[From the accompanying letter:]

Recognising that version "V2" of the Vancouver packet protocol can not meet the identification requirements stipulated until an updated version is released, the Department is prepared to authorise use of "V2" until 31 March 1987. It is anticipated that version "V3" will be available by this time and it is understood that "V3" will fully comply with the identification requirements.

From The Sydney Amateur Digital Communications Group (SADCG)

CLUBS ARE TRUMP

Packet of Long Island (POLI) is exploring the possibility of helping the New York Institute of Technology (NYIT) test student-developed networking hardware and software. NYIT is where POLI holds their monthly meetings.

The **Chicago Area Packet Radio Association** (CAPRA) continues work on networking software and hardware. At the moment, CAPRA is testing KA9Q's networking software for the IBM PC, with an eye toward getting it to run on CAPRA's 68000-based network controller.

NEW SOFTWARE FROM WA7MBL

Version 2.0 of YAPP, a packet terminal program for the IBM PC, is now available. It can be downloaded from Hamnet on Compuserve or is available from a number of landline BBSs. A copy is also available from:

Wes Morris, K7PYK
7422 E McKinley
Scottsdale, AZ 85257

Send either $5 or a blank disk plus stamped, addressed return mailer. Be sure to request YAPP.

Wes should be mailing an update to those of you who have contributed for an earlier version. If you feel you have a copy coming and don't receive it, please send Wes a note and he will update his records.

Version 3.11 of the WA7MBL BBS has being sent to the following beta testers: W2JUP, W9ZRX, K1OJH, AA4RE, KØGUZ, W3IWI & WB2MNF.

If no major bugs are found, v3.11 will be ready for distribution as soon as the documentation is cleaned up. If you wish to receive a copy when it is ready, send your request to Wes as above and he will ship as soon as it is released.

From Jeff Jacobson, WA7MBL, via DRNET

VOICE SYNTHESIZERS AND PACKET

[From an article by Art Human, W2HQQ, appearing in the POLI Parrot newsletter.]

Packet radio is enjoying enormous growth these days, but there is a group within our ham fraternity who cannot presently operate packet or other digital communications modes. I'm talking about the large number of blind radio amateurs. K1PXE, Pete can operate, but he has a Braille printer and several other devices beyond the price limits of most blind hams. WA1PDK, Larry of Stamford, Connecticut told me that one of his friends, N2AAM was working packet with difficulty with the aid of a Votrax text-to-voice converter. I saw a demonstration of this device at a computer club a few years ago; it connects to the RS-232-C output of a computer and attempts ASCII text-to-voice translation.

I believe that VOTRAX has since discontinued the unit I saw demonstrated, but there are two new models; one for the Apple II computer for $180 and another for the IBM PC and clones for $250. Both of these are plug-in cards and although they probably work better than the original Votrax, they use software for text-to-speech conversion. This means that you must use the computer the unit was designed for, and the computer is tied up doing the voice processing.

Both the original Votrax unit and the new devices generate speech using phonemes and allophones. A phoneme is a piece of sound, which can be put together with other pieces to form words. Allophones are variations of phonemes. The "p" in the word "pin" and the "p" in the word "spin" are different allophones of the same phoneme. Understandable speech can be produced using about 64 different allophones.

The original Votrax used the SC-01 CMOS generator chip, and the new ones use the SC-02. The new chip adds 32 inflections, 8 inflection slopes, 16 rates and 18 format transitions. I'm sure it sounds less like the original mechanical "computer speech" of the first unit, but as I said, these new devices require that you have software running on your computer, so they aren't really the answer to the "blind ham on packet" question.

IBM has come out with a text-to-speech software package with a cost of about $1500. Again, it ties up the computer but probably does a very good job.

Why this bother about the phoneme method and mechanical speech quality when our new repeater controllers have voice synthesizers that have excellent sound? The repeater speech synthesizers have a limited vocabulary. They can speak the preprogrammed messages required for repeater use, with some variations created by stringing words and phrases together, but they can't translate text to speech. You might actually consider the repeater controller more as a random-access digital voice recorder than a speech synthesizer; a typical device of this type has a vocabulary of about 140 words stored in 16 kbytes of EPROM. Many megabytes of memory would be required to store a small dictionary of 10,000 words. So the only viable technique for the present is allophone speech synthesis.

Radio Shack has been selling an allophone speech synthesizer since last spring. Their catalog number 276-1784 ($12.95) is actually a General Instrument SP-0256 chip. This summer they added the GI S256-AL2 Code-to-Speech Processor chip as their #276-1786 ($16.95). The S256-AL2 converts ASCII text to the control codes required by the SP-0256 for allophone speech synthesis. It is a complete microcontroller with 16 kbytes of ROM; however, the Radio Shack documentation warns of only "85% accuracy."

The SP-0256 requires a 3.120-MHz crystal; I had a 3.02-MHz crystal in my junk box. [A 3.58-MHz color-burst crystal also will work. Ed] The S256-AL2 requires a 10-MHz crystal. These crystals may be special ordered from Radio Shack, but their order delay time may be quite long. Most computer hobbyists should have both a 10-MHz and a color-burst crystal in their junk boxes. The rest of the components required are not difficult to locate, although they may not be available at Radio Shack.

Radio Shack supplies applications notes with the chips, including a few sample circuits. The unit I built from these chips has a 2-kbyte buffer and DIP switches for various baud rates and word-length/parity options. The text-to-speech micro is a well thought-out design. I wire-wrapped the unit on a prototype card and it worked from first power up (after installing the last forgotten jumper).

Does it Work?

It does work, no question about that. The speech generated sounds like mechanical "computer speech," almost identical to the Votrax unit of several years ago. The Votrax had a pitch control; substituting an adjustable TTL oscillator for the crystal in the Radio Shack circuit would probably let you vary the pitch somewhat. The audio sounds a bit distorted; the circuit uses an LM386 audio amplifier in its maximum-gain configuration. Using a different audio amplifier might get rid of some of the distortion, but it may be the best the speech chip can offer. It's still much better than any of TI's Speak and Spell devices and I would not consider it a limiting factor.

What Does "85% Accuracy" Mean?

I mentioned that the Radio Shack documentation mentions "85% accuracy." This means that a small microcontroller with 16 kbytes of ROM can't do everything. The 26 letters of the alphabet are handled by look-up tables, as are the numbers 0-9. This is because the spoken letters and numbers are really words, just like any other words, composed of several allophones. Shorter single-syllable words are also handled by look-up tables. The look-up tables, operating system overhead, adjustable baud rates and other parameters eat up a lot of the memory space. The remainder of the memory is used for the rules of pronunciation and interpretation used to create longer words. The English language is very complex, and the rules have many exceptions. The problem is that there isn't enough memory to handle all possible rules, much less any room for exceptions. If there was more memory available the accuracy would be better.

For example, when the device speaks the word "July," the ly is sounded the same as it is in truly or purely. To make July sound right would require an exception to the pronunciation rules. There's no room in memory for this exception, so July sounds like Julie. If the word is used in a sentence it shouldn't be too difficult to figure out, but it's confusing.

Mathematicians and speech people must have had lots of fun trying to figure out how to get the highest possible accuracy into the limited space. I think the chips do a good job, considering the simplicity of the circuit. The two main questions are: "Is it good enough?" and "What can be done to improve its usefulness?" As far as the first question is concerned, I do not believe this circuit is acceptable at present for Amateur Radio use.

Some Needed Improvements

Since this is a closed microcontroller system, nothing can be done internally to improve it. The data sheet says that more ROMs can be added to create specific word dictionaries and exceptions to the rules. General Instrument will only make information about connecting more ROMs available to firms that will be buying their voice development systems for this chip series, however.

For Amateur Radio use the first solution needed is a way to handle call letters. At present the "word" KA2EYW-1 would result in speech for the letters KA as a word, followed by the number 2 as a word, followed by an attempt to say EYW as a word and ended with the number 1. It's good for laughs and makes you wonder if Victor Borge is back at work with his old "sounds for punctuation marks" routine, but no good for on-the-air use.

To solve this problem (and a few others) a "black box" in the RS-232-C data stream between a source and the synthesizer is needed. The black box would include an 8-kbyte buffer and a control and decoding arrangement. When the decoder found a word with an embedded number it would insert spaces between all characters in that word. Call signs would then be spelled out, letter by letter, by the speech synthesizer.

Another use for the black box would be inserting longer pauses between words. Presently, the time between words is not controllable, and longer pauses between some words would make it easier to understand them. A microcontroller could easily handle this as long as its storage buffer didn't overflow. The buffer might overflow while a packet station was accessing long files on a BBS, but probably not during a packet or RTTY QSO.

Another useful function would be a repeat key. Pressing this key would cause the device to back up and repeat one or more words that were difficult the first time.

A repeat and spell key would place spaces between the letters in previous words that were unclear even after a repeat. A spell only switch also seems reasonable.

Why all this fuss and bother? This device could be a low-cost solution to an intriguing problem. Any other solution would cost much more. The device doesn't require any computer, much less a specific one. For demonstration purposes a VIC20 with Hamsoft cartridge, without video modulator or TV set, becomes a simple 300-baud ASCII source.

Quo Vadis (Where are we going?)

The improvements suggested in this article are ideas that I came up with. WA1PDK is presently evaluating the device, and he may decide that the device is not worth investing any more time and energy on, but with the total cost so far only $35, I felt I had to at least try it.

Any suggestions I get from Larry will be incorporated into the "black box" I plan to design. This will take a bit longer than just following the Radio Shack schematic, but stay tuned.

MORE ON PICTURES VIA PACKET

In Gateway, Volume 3, Number 8, we reported on the use of the TELIPAK system to send pictures via packet. Glynn E. Rogers, K4ABT, wrote to inform us that his system, called Digital Image Transfer (DIT), was first demonstrated via packet at the SouthNet packet conference in Gainesville, Florida in November of 1984.

SADCG RELEASES NEW VADCG TNC SOFTWARE

The November/December 1986 issue of The Australian Packeteer, the newsletter of the Sydney Amateur Digital Communications Group (SADCG), contains the following announcement:

Thanks to the untiring and exhaustive efforts of Les, VK2KYJ, a standard version of AX.25 software for VADCG TNCs has been produced.

This version is based on the VADCG V2 protocol AX.3 TIP and the AMRAD Version 5.5 G/J AX.25 module. Extensions have been made in the following areas:

- up to 8 digipeaters may be chained in one connect string with a command of the form:

#co VK2XXX via VK2RMB VK2RPH.... (up to 8 digis)

- The command prompt for AX,25 is a hash symbol: # to differentiate from the V2 prompt: *.
- The TNC defaults to digipeating ON.

The AX.25/AX.3 module is the only version that will be supported in the future by the SADCG.

Contact SADCG for further information:

SADCG
P.O. Box 231
French's Forest, NSW
AUSTRALIA 2086

JAPAN'S PARNET TNC

[This brief description of the Japanese PARNET TNC is also from The Australian Packeteer. It is based on articles in the Japanese magazines Ham Journal, number 43 and CQ ham radio, Oct. 85.]

PARNET (Packet Amateur Radio Network) is a group of five hams (JA1MIR, JH1UWU, JE1WAZ, JI1BXM and

JA3FGN) interested in using amateur radio for data communications in general, and computer networking in particular. They decided they wanted a TNC that 1) used only components readily available in Japan and 2) required no adjustment. To meet these requirements, they built their own TNC using an 8273 HDLC-controller IC and 7910 modem IC. They further decided to support the AX.25 V2 protocol, but with the following restrictions:

1) Because digipeating is illegal in Japan, only a single digipeat is supported (for closed-loop testing).
2) The TNC requires an "ack" after each transmitted frame (rather than allowing up to seven outstanding frames, as AX.25 permits) because of the crowded, jamming-prone Japanese bands.

There are several commands that can be used to control the TNC's operating modes, including the ability to monitor received frames, displaying the address, control and information fields of the frames. Both conversational and binary (file-transfer) modes are supported, and the timing parameters of the TNC are settable by user commands.

REPRODUCTION OF GATEWAY MATERIAL

Material may be excerpted from Gateway without prior permission, provided that the original contributor is credited and Gateway is identified as the source.

Gateway: The ARRL Packet Radio Newsletter is published by the:

American Radio Relay League
225 Main Street
Newington, CT 06111

Jon Bloom, KE3Z
Bruce Hale, KB1MW
Editors

Larry E. Price, W4RA
President

David Sumner, K1ZZ
Executive Vice President

Subscriptions are available to ARRL members and nonmembers, at the following rates:

For 25 issues of Gateway:
In the US
ARRL Member $6.00
Nonmember $9.00
In the US, Canada and Mexico by First Class mail
ARRL Member $11.00
Nonmember $14.00
Elsewhere by Airmail
ARRL Member $14.00
Nonmember $17.00

Prices are subject to change without notice. Remittance must be in US funds, and checks must be drawn on a bank in the US.

Gateway: The ARRL
Packet Radio Newsletter
American Radio Relay League
225 Main St.
Newington, CT 06111

Nonprofit Organization
U.S. Postage
PAID
Hartford, Conn.
Permit No. 2929

7045536 N8GNJ P 24
Steven K Stroh
917 Salem Ave
Elyria OH 44035-1825